Lecture 6
Block Ciphers

CS3690 Network Security
Summer Quarter, 2000
C. Irvine

Objectives

Block Ciphers

- Stream ciphers
  - ★ Encrypt one bit at a time and include Vigenere and Vernam as examples
- Block Cipher
  - ★ Encrypt blocks of bits, often 64 bits, at a time
  - ★ A non-singular transformation is used to encipher the plain text

Reversible		Irreversible	
plaintext	ciphertext	plaintext	ciphertext
00	11	00	11
01	10	10	10
10	00	10	01
11	01	11	01

  - ★ Of course, transformation needs to use a larger block size.

Block Sizes

- With a large block size the cipher is not vulnerable to the cryptanalysis that makes ordinary substitution ciphers so vulnerable
- We cannot use an arbitrary reversible substitution cipher for large block sizes because the substitution table is, in effect, the "key" for the algorithm. As the block size increases, the size of the substitution table will become enormous. It will be . For a 64-bit blocksize the "key" becomes 2^{70} bits.

Feistel Ciphers

- Feistel: approximation to an ideal block cipher may be "good enough."
  - ★ only needs to be computationally secure, not perfectly secure.
- Feistel suggested using two simple but different basic ciphers serially to produce what is effectively a stronger result.
- Shannon: ciphertext must be independent of the key used
- Permutations provide diffusion
  - ★ The objective is to obfuscate the relationship between he plaintext and the cipher text so that the key cannot be deduced.
  - ★ structure of the plaintext is diffused over the ciphertext. A plaintext digit affects many ciphertextdigits.
  - ★ Diffusion is more than moving blocks of strings around; bits will be moved.
- Substitutions provide confusion
  - ★ Obscure relationship between statistics of the ciphertext and of key.
  - ★ Make the way in which the key is used extremely complex by using an involved substitution algorithm.

Feistel Network

- Description of a round
  - ★ Plaintext is split into two blocks of equal length.
  - ★ A substitution is applied to the Left half
  - ★ A round function is applied to the Right half
  - ★ XOR performed
  - ★ Permutation then start another round.
- Design Features
  - ★ Block size - make it large
  - ★ Key size - make it large, but practical for speed
  - ★ Number of rounds - make it large enough
  - ★ Subkey generation algorithm - complexity is good
  - ★ Round function - complexity is good
- Additional concerns
  - ★ hardware implementation
  - ★ easy to study effectiveness

History of DES

- 1972:NBS issued a call for proposals:
  - Must provide high level of security.
  - Must be completely specified and easy to understand.
  - The algorithm itself must provide the security.
  - Must be available to all users.
  - Must be adaptable for use in diverse applications.
  - Must be economical to implement in electronic devices.
  - Must be efficient.
  - Must be able to be validated.
  - Must be exportable.
- 1974: IBM responded with “Lucifer” (renamed - DEA).
- 1976: DES officially adopted.

Overview of DES

- Combination of:
  - Substitution technique (for confusion).
  - Transposition technique (for diffusion).
  - These two techniques are repeated for 16 cycles one on top of the other.
  - Plaintext is encrypted in blocks of 64 bits.
  - Keys are 64 bits long (only 56 are really needed).
  - Uses only standard arithmetic and logical operations on up to 64 bit numbers.
- Four Modes of Operation
  - ECB -Electronic Code Book
  - CBC - Cipher Block Chaining
  - OFB - Output Feedback
  - CFB - Cipher Feedback

ECB Mode of DES (Native Mode)

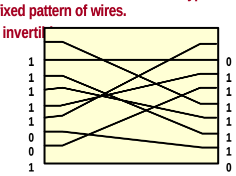
- Encrypt Mode
  - Data input into ECB must have a block size of 64 bits (8 bytes) long.
  - Key must also be 64 bits (only 56 are used).
- Decrypt Mode
  - Ciphertext block is input
  - Key is the same used for encryption.
- Bit Sensitivity
  - If only one bit of either the input plaintext or key is changed, the output block will be completely different.
- Basic Building Blocks: two distinct algorithms
  - Crypting algorithm
  - Key scheduler
- Each composed of subprograms
  - Permutation Boxes (P-boxes)
  - Substitution Boxes (S-boxes)

Internal Overview

- The message block of 64 bits
  - divided into submessages (sub blocks)
  - each submessage is input to a transformation function

DES - Permutation (P-Box)

- Permutation moves bits
- The figure below shows the permutation of the EBCDIC character “9” (binary 11111001). The P-box has transformed it into (01111110) the EBCDIC character “=“
  - The diffusion process of the P-Box has disguised the information.
  - If we consider the P-Box to be a mini-encryption algorithm then the key is the fixed pattern of wires.
  - Process is inverti



Exclusive OR (XOR) Operation

- Can be viewed also as a mini-encryption algorithm
- Also known as “addition modulo two”
- Process is invertible

DES - Substitution Box

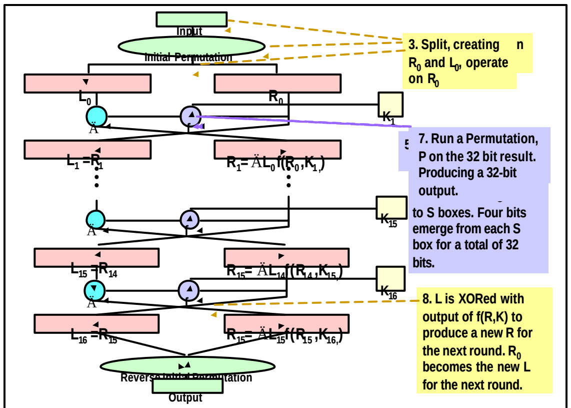
- Introduces confusion and non-linearity to DES
- Interpret bits as numbers
- One number replaced by another from a table
  - ★ table has values ranging from 0 (0000) to 15 (1111)
  - ★ duplications among the elements
- Takes 6-bit input and returns 4-bit output
  - First and last bits choose row into S-box substitution table.
  - The middle four bits chooses the column
  - The table returns a four bit number
- Central to the algorithm's secrecy

S Boxes

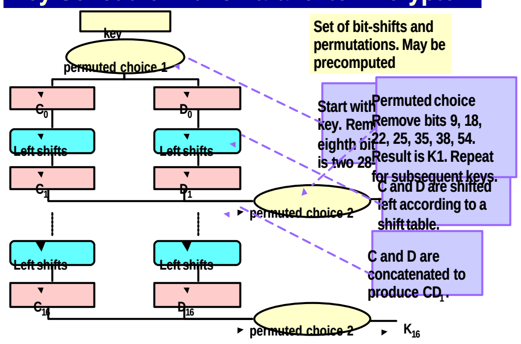
- Example S-Box transformation
  - ★ input decimal 43 = 101011 binary
  - ★ choose first and last bits 11 = decimal 3 = row 3
  - ★ middle four bits 0101 = 5 decimal = column 5
  - ★ output is 9 decimal = 1001 binary

Sample S-Box

	Column Number															
Row No	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13



Key Scheduler Runs Parallel to Encryptor



Avalanche Effect

- Highly desireable in a cryptographic algorithm
  - ★ one bit change in either the key or the input block will result in a significant change in ciphertext
  - ★ each input bit has an effect on each output bit
  - ★ experimental example (your results will vary slightly)
    - take 2 plaintexts that differ by only one bit
    - after 2 rounds outputs differ by 21 bits
    - after 16 rounds outputs differ by 34
    - take 2 keys that differ by only one bit
    - after 2 rounds outputs differ by 14 bits
    - after 16 rounds the output differs by 35 bits

DES Criticisms

- Not Strong Enough Anymore
- Are 16 iterations enough?
- Did NSA fool around with the S Boxes?
- Did NSA put a trap door into DES?
- Weaknesses of the DES
  - ★ Weak keys (e.g. all zeros or all ones).
  - ★ Semi-Weak keys (2 separate keys can decrypt the same message).
  - ★ The same DES algorithm is used!
  - ★ The keys are used in reverse order
  - ★ Key length inadequate
- DES intentionally designed for hardware implementation
  - ★ permutations are fast in hardware but slow in software

Weaknesses of ECB Mode

- ECB Mode encrypts a 64-bit block independently of all other 64-bit blocks
  - ★ Given the same key, identical plaintext will encrypt the same way
- Data compression prior to ECB can help (as with any mode)
- Fixed block size of 64 bits
  - ★ incomplete block must be padded
- ECB error propagation
  - ★ Since each block is independent of previous blocks only those in error need be resent

Other Modes of DES: What For?

- ECB Mode has obvious weaknesses
- CBC mode reduces problem of repeated plaintext
  - ★ Makes replay of existing ciphertext more difficult
- OFB Used for streaming data
  - ★ When you have continuous data, e.g., realtime
- CFB Used for streaming data
  - ★ Good for realtime

Cipher Block Chaining Mode

- The encryption of each block depends upon the encryption of the previous block
  1. C1 = E(B1)
  2. C2 = E(E(B1) Ä B2) = E(C1 Ä B2)
  3. C3 = E(E(E(B1) Ä B2) Ä B3) = E(C2 Ä B3)
  4. etc.
- Must use an initializing vector (IV)
- The IV may be viewed as a second key
- CBC Mode Error Propagation
  - ★ A single bit error affects two blocks

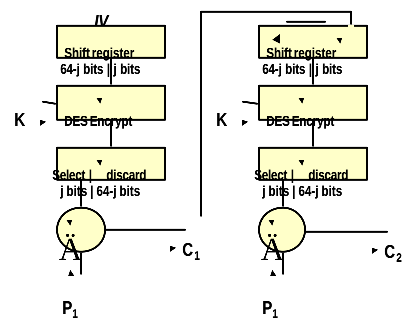
DES - Feedback Modes

- 64-bit block now replaced by a K-bit block
- Possible to encrypt data of any length
- A 1-bit feedback would require 64 cycles to encrypt 64 bits of data
- Ability to encrypt on a character-by-character basis
- Trade-off: Speed vs. Flexibility of block size
- Used for Message Authentication Codes (MAC)
- Frequently referred to as message digests

Cipher Feedback Mode

- On encryption, the ciphertext, rather than the output from DES is fed back
- Can affect two blocks, for same reasoning as ECB
- Used for communications
- Transparency to protocols
- Greater security
- Used for encrypting fields within a record
- flexibility in block size

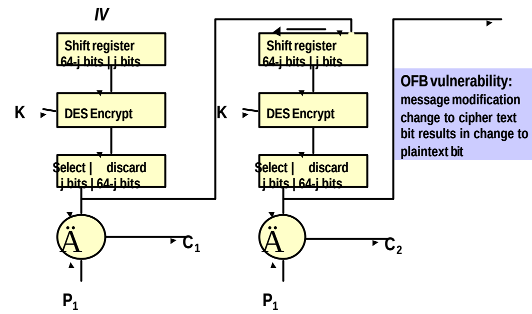
J-bit Cipher Feedback mode



Output Feedback Mode

- DES in OFB can be used as random number generator
- IV is the seed
- DES is used in encrypt mode for both encryption and decryption of text since a reverse key schedule cannot be used because of the XOR operation
- OFB often used as a random number generator
- OFB Error propagation
  - Only a single bit is affected

Output Feedback mode



International Data Encryption Algorithm

- Overview
  - Operates on 64-bit plaintext block.
  - Uses 128 bit key.
  - Same algorithm is used for encryption and decryption (like DES).
  - Considered by some to be superior to DES
- General Description
  - 64 bit input block is divided into four 16 bit blocks: X1, X2, X3, and X4 which become the input blocks to the first round of the algorithm.
  - In each of the eight total rounds, the four sub-blocks are XORed, added, and multiplied with one another and with six 16 bit sub-blocks of key material.
  - Between each round the second and third sub-blocks are swapped

Speed of IDEA

- Software implementation speeds are comparable with those for DES.
- Hardware implementations are just slightly faster.
- The algorithm was designed to achieve high data throughput for use in real-time communications system.